EECS 213: Homework 3

Memory and Cache

Spring 2007

Important Dates

Out: May 10, 2007.

Due: May 18, 2007 (11:59PM)

Submitting your homework: Please use the course submission site. There is a link to it from the class site. **Submit only ASCII text files.**

To be done individually.

1. Consider a processor which uses 16 bit addresses and can address $2^{16} = 64Kbytes$ of memory. Suppose that it has one level of cache. As in Figure 6.25 of your textbook, the address is split into a t bit tag, an s bit set index, and a b bit block offset. The cache consists of 1024 bytes, with a block size of 32 bytes. Answer each of the following for direct-mapped, 4-way set associative, and fully associative versions of the cache.

   (a) How many cache lines are there?

   (b) What is b?

   (c) What is s?

   (d) What is t?

2. For the cache in problem 1, draw the cache given it is structured as follows. You can elide replicated components, but annotate your drawing with how many components there are.

   (a) Direct-mapped

   (b) 4-way set associative

   (c) Fully associative

3. Our company wants to optimize the performance of the following code

```
void vector_add(int n, int *a, int *b, int *c)
{
  int i;
```

```
  for (i=0;i<n;i++) {
    c[i]=a[i]+b[i];
  }
}
```

to run on the same processor and cache as described in problem 1. The cache is write-back, write-allocate, and has an LRU replacement policy. Integers are 32 bits.

(a) Suppose the cache is direct mapped. Let $n = 2048$, $a = 0x4000$, $b = 0x8000$, $c = 0xc000$. On average, how many times per loop iteration will you load a cache block from main memory? How many times per loop iteration will you flush a cache block back to main memory?

(b) What is the minimum degree of associativity (i.e., the n in n-way) that the cache needs to reduce the answers in (a) to 0.375 cache blocks read per iteration and 0.125 cache blocks written per iteration?

(c) While were all fired up to buy ultra-cool mega-associative cache hardware (which comes only in black), a smart alec programmer claims that we can get the same effect by having $a = 0x4000$, $b = 0x8020$, and $c = 0xc040$. Is he right? Why or why not?